

## 2次元展開にもとづく被服の3次元表示

濱 田 渉\*・成 久 洋 之\*\*

\*岡山理科大学工学研究科修士課程電子工学専攻

\*\*岡山大学理科大学工学部情報工学科

(1992年9月30日受理)

### 1. はじめに

被服の製作はその多くの工程が人間の手と経験にもとづく試行錯誤を主とする作業形態で行われている。これは、被服を着用する対象である人間の体型が、展開不可能な立体形状である事によるものである。そのため、被服は人体の体型をデフォルメしたうえで、その材料である布地の柔軟性や屈曲性を利用して設計・製作される。このように被服の平面時の形状と立体に編成した形状の間の関連性を明らかにしておく事は非常に重要である。

その見地から、昨年我々は立体裁断の方法にもとづき、被服を着用した形状から展開図をもとめる事によって、立体を構成するための面の性質について研究をおこなった。

本来、被服設計は人体の形状値（肩幅や裾丈等）に基づき裁断されるものであるが、人体表面に適合した着用形態とその2次元展開との対応関係が設計の条件となるものである。

そこで、本研究ではその前年度の研究にもとづき、展開を行って得られた型紙のデータを元に、被服のパターンの立体表示を行うための手法を検討するものである。

### 2. システムの概要

被服の立体形状をコンピュータの内部モデルとして表現する方法としてはコンピュータ内部で扱いやすいデータとする事を目的に、その形状を必要な精度をもって近似しうる小さな平面の集合体として扱う事ができる。

本研究では被服を着用した状態を3次元の構造のデータとして扱うが、その構造の単位となるデータをパッチと呼ばれる四角形の小平面の集合体によって構成する。これは立体のモデルを展開する際に縦方向の連続性と、横方向の連続性を区別する必要があるためである。一般的には人体に対する適合度の高い服ほど型紙には多くの断裂が必要となるが、断裂の多い服ではその縫合線がデザインを損う。そのため、被服のデザインにおいて断裂は、縦方向に数本のみに限定される事が多い。

本研究では昨年製作した立体裁断の手法にもとづいて被服の展開を行うシステムの結果を利用し、その拡張としてスキャンラインアルゴリズムを応用して被服の立体表示を行うシステムを作成した。

まず、立体形状のモデルのデータを読み込み、その型紙を作成する。この時作成される型紙には2種類あり、面の形状の保存を重視した展開方法と、布の柔軟性、屈曲性を利用した展開方法との2通りの物を作成できる。本システムでは、その両方に対して立体表示を行う事ができるが、実際には前者の方法による型紙は被服製作に利用するための物としては不適当なため、主に後者の手法で製作された型紙を利用することになると思われる。

続いて、型紙に対する布地となるマッピングデータを表示し、型紙に対するフィッティングを行う。

それらの準備・設定が終了した後にスキャンライン法にもとづいて被服の立体表示を行う。

### 3. データ構造

#### 3.1 データの扱い（構造）についての概要

本システムでは立体をパッチとよばれる4つの端点からなる小平面による構造体としてとらえている。そのため、立体形状データを表現するには、端点の座標値のデータとその連結状態をあらわすデータの2つにわけて考えることができる。そこで、本システムでは立体をあらわす各端点のデータとしてX, Y, Z軸それぞれの要素の値をもつaxis型のデータを登録し、これに番号を与える。これにパッチの各端点の接続を表すために、登録した単点の番号をパッチの4つの単点の位置を表すデータとして、その連結の関係をあらわすquadrangle型を用意し、それにもとづいて立体や平面の構成をあらわす事を考えた。

#### 3.2 表示モデル関係

##### axis\_array 型

本システムでは端点のデータはx, y, z軸それぞれの座標軸の成分をdouble型のデータとしてまとめたaxis型のデータとして扱う。また、同じ位置を表す端点を速やかに判断する必要性などから、端点のデータの番号付けを行い、接続している（重複している）点を表すデータには同じ端点番号が割り当てられる様にした。また、このことによって同じ座標値データに対し異なった単点番号を与える事によって、パッチの分断の際の分離線の指示を与える処理を行っている。このとき、axis型のデータと単点番号を対応させるためにaxis型のデータの配列として、axis\_array型のデータとして記録される。

##### quad\_array 型

本システムではモデルをパッチと呼ばれる四角形の小平面の集合によって構成された立体として扱う。パッチの小平面はquadrangle型のデータとして、パッチの4つの端点の座標番号の集合として表される。本システムでは立体のモデルはパッチの集合として表されるので、1つの立体をあらわすために必要なパッチをまとめて記録・管理するためのデータ型がquad\_array型であり、quadrangle型をデータを配列として整列したものである。

### model 型

model 型は、面とその端点によって構成される立体を扱うためのタグとなるデータである。モデルを構成する面データと端点データの実体は他のデータに保持される (quad\_array 型, axis\_array 型) が、そのデータへのポインタが model 型のデータでまとめられることにより結合され、1つの立体を表現する事になる。また、立体表示する際の角度や、ワールド座標内でのオフセット値(立体モデル表示時)、展開図のオフセット値、拡大倍率等のデータを持つ。これは立体表示の際の表示モデルのパーツ化をサポートするための物である。また、モデルデータはそのデータの種別を ID として保持している。これは model データを多義的に扱い、関数の共用化をはかり汎用性を高めるための工夫である。また、memory 型のリンクチェーンに model 型のデータが登録されている場合、それは展開やマッピングファイルの情報を含まないデータ (未展開のデータ) として扱われる。

### map.model 型

今回行った立体表示においては、被服の立体形状とその展開図の関係を対応付ける必要があった。そのための立体形状のデータと展開図のデータ、そして展開図に対応して、表示するモデルに対するマッピング情報のハンドルデータを統合するためのデータがその型である。

### assemble 型

今回のシステムでは、1つのモデルを表すデータを複数個関連づける事によって、1つの処理対象を表現する。そのため1件のモデルデータは、model 型または map.model 型のデータによって定義される。この2種類のデータを assemble 型のデータとして統合する事により同じ型のデータとして扱う事ができるようになる。これは、被服の各部位の形状データを表す2種類のデータを抽象的に1つのデータとして扱うためのハンドルとなるデータである。

### memory 型

今回のシステムでは、1つのモデルを表すデータ複数個関連づける事によって、1つの処理対象を表現する。そのため1件のモデルデータは、model 型, map.model 型のデータによって定義され、そのそれぞれを assemble 型に統合する事により抽象化を行い、それを memory 型のデータに登録し、それをリスト構造状に連結する事によって1つの完成体を表す事にする。

## 4. 描画アルゴリズム

コンピュータの内部モデルを立体形状として、画面上に表示するためのアルゴリズムとしては、レイトレーシング、Zバッファ法、スキャンライン法、Z軸ソートによるポリゴン表示等の手段も考えられる。本システムでは、開発を行ったコンピュータの処理能力を考慮してスキャンラインアルゴリズムを利用することにした。

スキャンライン法は、Zバッファ法の変形アルゴリズムであるが、Zバッファ法最大の問題であるワークエリアの消費を小規模なものにすることができる。その反面、データの評価回数が増えるため、計算量は増加する。しかし、複数回の光の経路変化要素以外についてはレイトレーシング法とほぼ同等の表現を行うことができる。また、計算量自体もレイトレーシング法に比較すると、かなり少なくすむ利点がある。

#### 4.1 スキャンラインアルゴリズムについて

近年のコンピュータのディスプレイはそのほとんど全てがラスタスキャン方式による表示となり、コンピュータは画面の解像度の分だけ、色情報を格納する領域を持つようになった。このことは、ある物体を表現するには、その物体が画面上に占める範囲の画素に対応するメモリー領域に物体の色情報を格納していることで表現ができるという事である。

スキャンラインアルゴリズムでは、いったん透視変換によって視野を表す画面上に描画

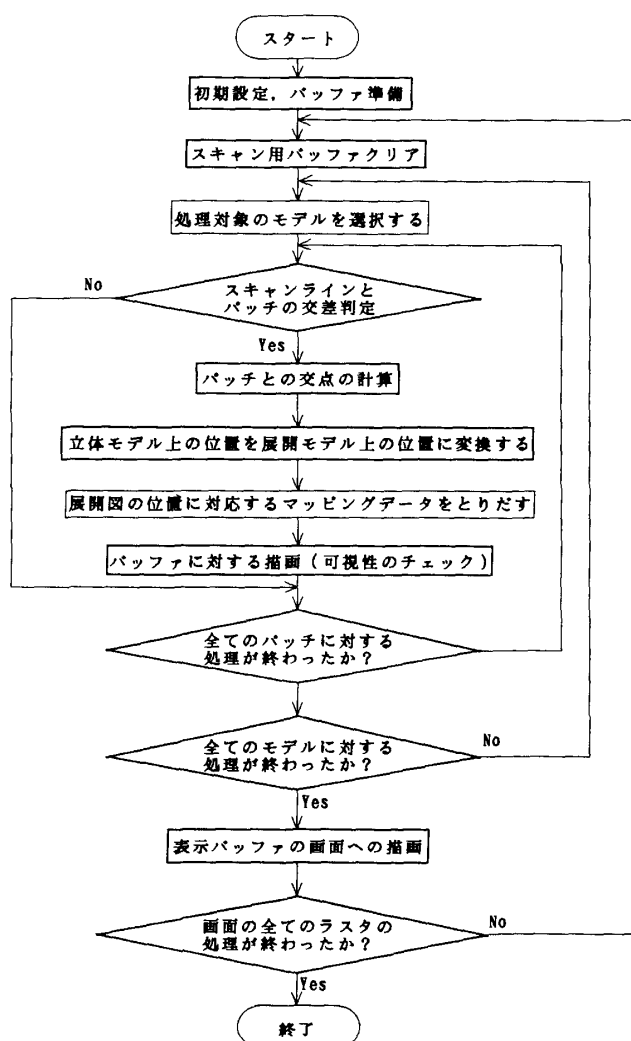


図1 スキャンライン部のフロー

対象物体が占める領域を求める際に、視野平面を横、または縦の走査線との交点を求めることにより行う。このため、一度に処理できるのは横、または縦の1ライン（スキャンライン）分のみであり、このスキャンラインを縦、または横方向にスイープすることにより、全画面の表示を行う。スキャンライン法にもとづく今回の表示アルゴリズムのフローチャートを図1に示す。

#### 4.1.1 スキャンライン法による表示手順・概略

手順としては、まず立体モデルの座標データを透視変換する。この座標をもとにパッチのスキャンラインとの交差をチェックする。処理対象となっているパッチがスキャンラインと交差していれば、そのパッチを2つの三角形に分割し、スキャンラインとの交差位置を計算する。さらに、スキャンライン上での可視性を検証する、可視な位置であれば画面バッファに対し描画を行う。このとき、モデルをマッピングするにあいには立体モデル中でのパッチ内の位置から、展開モデルの対応するパッチの位置を算出し、その位置に対して与えられた色データをマッピングファイルからとりだし、色情報として設定する。

#### 4.1.2 スキャンラインの走査について

スキャンラインを画面の画素の列と考え、画面は左手系の立体座標を透視変換するスクリーンの座標に対応するとする。このとき、スクリーンの画面は横方向（右方向）にX軸、縦方向（上方向）にY軸が対応する事になる。またこのとき、画面の中央を透視スクリーンの原点とする。

この場合、スキャンラインの走査方向においてはY軸成分の変化はない事から、スキャンライン上の点については、X軸の位置のみが処理対象になる。また、処理対象が平面なので、Z軸成分は二点間の線形補間を行う事によって求めることができる。

### 4.2 スキャンライン部におけるパッチの表現

#### 4.2.1 パッチの三角形分解について

スキャンライン法において、パッチとスキャンラインとの交点を求めようとする場合には、スキャンラインと交差しているパッチをいったん2つの三角形に分割し、それぞれの三角形の描画とする方が計算上都合がよい。

また、パッチが端点の位置データと、その関係のみによって定義されている為、パッチの面の状態には、2つの解釈が発生する事があり、適切な立体表示のためにはこの面の状態を一意に決定するために、任意のパッチ稜線を決定する規則を用意しておく必要がある事を示している。実際には、被服は人間の体を包むように製作されるため、単独のパッチが凹型になるような極端な形はあまりないと思われることから、今回はパッチの対角線の中点がモデルの原点より離れている方の対角線をもってパッチを2つの三角形として分割する規則を採用する事にした。

#### 4.2.2 スキャンラインと、分割三角形の交点計算

スキャンラインを Y-Z 平面と平行な面とする。この時、透視画面上の座標系が左手系であり、スキャンラインは画面の左右方向となる。これによって投影変換された分割三角形は、スキャンラインと 2 辺、または頂点で交わる。また 2 つの平面の交差する部分は必ず直線になる。そのため、端点間の補間には直線補間が使用できる。

そこで、この三角形とスキャンラインとの交点の座標は、図 5 のような場合を考えると、2 つの線分の位置関係からスキャンラインとの交点 ( $L_x$ ,  $L_y$ ,  $L_z$ ) は、

$$L_x = \frac{(X_1 - X_2) * (Y - Y_2)}{Y_1 - Y_2}$$

$$L_y = Y$$

$$L_z = \frac{(Z_1 - Z_2) * (Y - Y_2)}{Y_1 - Y_2}$$

(Y はスキャンラインの Y 座標)

の様を求めることができる。もう一方の交点の座標 ( $R_x$ ,  $R_y$ ,  $R_z$ ) も同様である。

#### 4.3 パッチ上の位置の変換計算

ここまでのフェイズで立体モデルの表面とスキャンラインとの交点が求められた。その

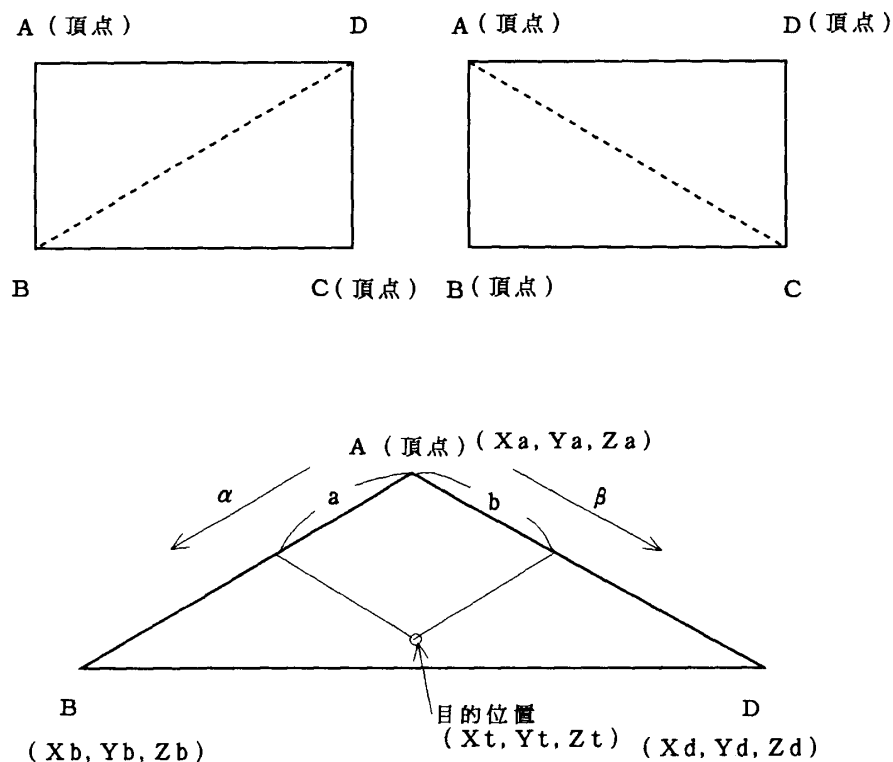


図2 パッチ内位置の変換

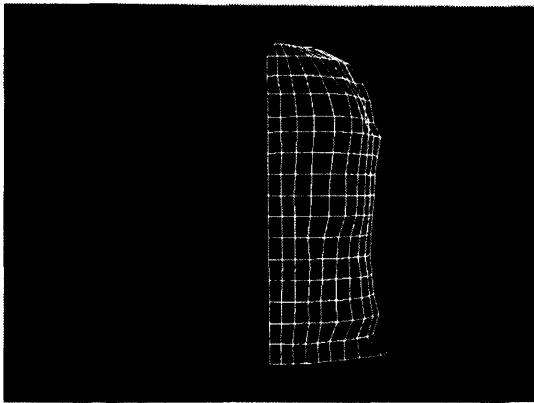


図3 立体モデル（正面図）

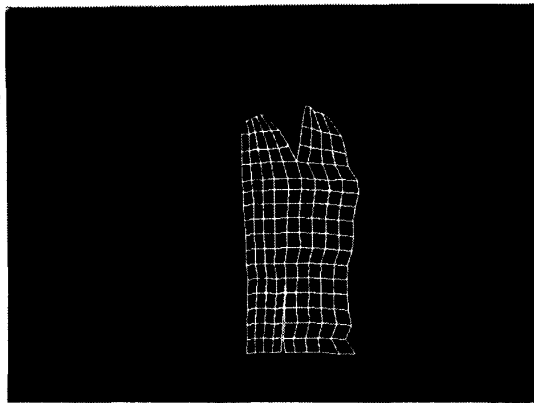


図4 立体モデル展開図（型紙）

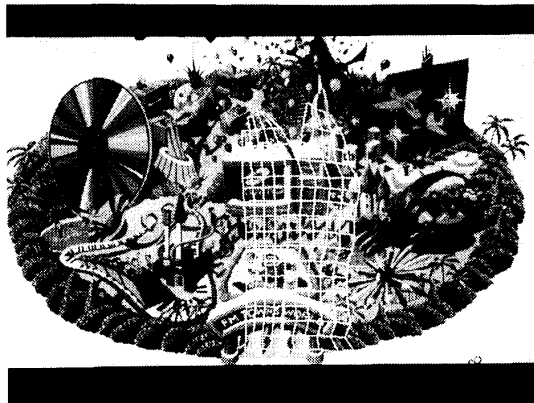


図5 マッピングデータとのフィッティング



図6 3次元表示例（正面図）

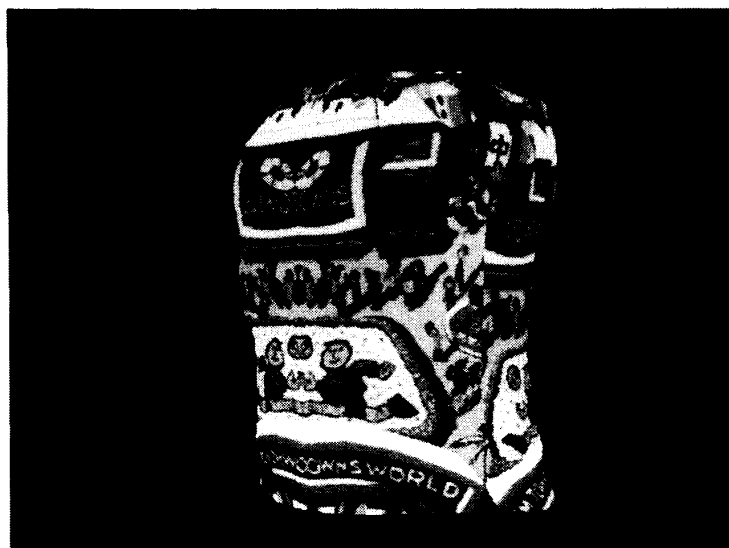


図7 完成体表示例（45度位置）

うえてその面の可視性を1ラインのZ軸バッファの情報をもとに判断する。そこで可視面と判断されたパッチの現在注目している位置と対応する型紙の模様の位置を求め、その生地の色データを取り出すのであるが、ここで問題となるのはパッチの変形である。

実用的な型紙として、布の変形を考慮に入れた型紙を利用できるようにしているため、単純なオフセット値等による方法は使用できない。そこで、パッチの各辺を単位ベクトルとし、その合成ベクトルによって目的位置を表すパラメータを求める事によって、パッチ内位置を表示する変換を行うことにした。その様子を図2に示す。

各三角形内の内部位置はパッチの分割線の端点でない頂点Aからの相対座標 ( $T_x$ ,  $T_y$ ) にいったん変換する。続いて頂点Aから三角形の各端点へのベクトルを求めこれらのベクトルをそれぞれ  $\alpha$ ,  $\beta$  とする。

ここで

$$\alpha = B - A = (\alpha_x, \alpha_y)$$

$$\beta = C - A = (\beta_x, \beta_y)$$

より、求めようとする点の座標を示すパラメータを ( $a$ ,  $b$ ) とすると、

$$a \cdot \alpha_x + b \cdot \beta_x = T_x$$

$$a \cdot \alpha_y + b \cdot \beta_y = T_y$$

の関係式があらわされる。この方程式をもとに Gauss-Jordan 法を用いて求めた解 ( $a$ ,  $b$ ) を変換のためのパラメータとして使用した。

求める展開モデルの該当する頂点とその両辺のデータを上の方程式に与えることによって、展開モデル (型紙) 上のそのパッチ内の対応する位置を計算することができる。

## 5. 結 果

これまでに示したようなシステムを作成し、これに立体形状のデータとして図3のようなモデルを半身のデータとして与えた。これを男子の体にフィットしたシャツの半身の前身ごろの形状とする。このモデルを展開することによって、このモデルを実現するための型紙として図4のような展開モデル (型紙) を得た。

続いて、このモデルに対する布地の模様として、立体の形状の把握しやすさを考慮して、図5のようなマッピングデータとのフィッティングを行った。

その結果、得られた立体の表示が図6 (正面図) である。

また、体幹部各部のモデルデータに対して同様の操作を行い、被服の完成体の様子を検討すべく、立体表示を行ったのが図7である。

## 6. 結 論

本研究は被服設計における生地のカットと人体にぴったり適合した服を着用した状態での3次元表示との対応関係を、個人の身体形状に適合した被服の立体カットとその2次元展開



に基づく3次元表示との関係に注目して検討したものである。その目的に沿って開発した表示システムによって得られた結果は、図3から図7において示した。

これらの結果においては立体の表面の傾向に基づいた模様の傾きの変化が発生している事が認められる。これは立体のモデルと、その2次元展開図の関係を対応づけられた事を意味している。

#### 参考文献

- 1) 成久洋之, 濱田 渉: "被服の2次元展開について", 岡山理科大学紀要第27号 Ap. 241—250, 岡山 (1992)
- 2) 小池千枝: "服飾造形論", 文化出版, 東京 (1981)

## On the 3-Dimensional representation from 2-Dimensional expanding of clothes

Wataru HAMADA\*, Hiroyuki NARIHISA\*\*

*\*Graduate School of Engineering*

*\*\*Faculty of Engineering,*

*Okayama University of Science,*

*Ridai-cho 1-1, Okayama 700, Japan*

(Received September 30, 1992)

Human body shape that is weared clothes is not able to expand so that the wear is made up on to try and error. Thus, the wear is designed and made up by using deformationn form, cloth's curve and distortion. To design and made up the wear, it is needed to be cleary definition of between wear's plane form and solid form. On the previous research, we studied the relations of clothes and solid model of human-body, concerning with draping which is one of the former techniques of clothes design. In this paper, we discuss the way of 3-Dimensional displaying by using the pattern which is got by 2-Dimensional expanding.